Software Tools

C, Unix (Linux), and tools

Unix/Linux

- Unix was born in the late 60's-early 70's out of the corpse of the failed Multix project.
- Unix was a far simpler system than Multix, but included many of the Multix innovations
 - Files were simply streams of bytes
 - And wherever possible, human readable
 - The OS was written in a high level language
 - The filesystem was hierarchical
 - Interactive shell same as the script interpreter

Unix Philosophy

- Programs should do one thing but do it well
- Programs should work together
- Programs should handle text streams
- It should be easy to write, test and deplay programs
- Should be self maintained (ie s/w should be maintained under Unix)

What does an OS Do

- File management
- Process management
- Memory management
- I/O management

Components of an OS

- Kernel
- System programs
 - Shell
 - Compiler
 - Utilities

File System

- Files are organized in directories, sub-directories etc
 - It is a tree like structure
- The *root* directory is /
- All file names are are either absolute or relative to the current directory (cwd or .)
- A file name is also known as *path*
- Your home directory is ~userid (e.g. ~minas)
- All *paths* are unique
 - But two filenames can have the same basename if they are in different directories.

File System

- The filesystem is also a namespace.
- We can put on it things that are not files
 - Devices
 - Kernel structs
 - Named pipes
 - Mountpoints

Process Management

- A process (task) is a running program
- The OS can run multiple processes at the same time
 - The processes share the CPU (even a single core CPU
 - This is done by switching between the processes several times a second.
 - If we are lucky while one process uses the CPU the other is doing I/O
- Each process has its own process ID (*pid*)

Devices

- Unix/Linux treats devices like files (we can open, read, write, fcntl, close them)
 - Use man -s 2 open
 - Or man -s 2 close
- They are in the dev directory
- The communication with devices is done in a way similar to files
- Devices in dev can appear or disappear when plugging/unplugging USB devices

Shell

- For reasons lost to history the program through which programmers interact with the system is called shell.
- The most popular is bash (Bourne Again Shell)
- Executes commands typed interactively
- Same commands can be used in batch mode
- Provides assistance when typing

Common Shell Commands

- List files: Is
 - -I long format
 - -a all files (including hidden files that start with a .)
 - -t sort by time
 - -r in reverse order
- Make a new directory: mkdir
- Change current directory: cd
 - Change to home: cd
 - Change to parent directory: cd ..

Common Shell Commands

- Print working directory: pwd
- Copy file: cp <oldfile> <newfile>
 - Or: cp <oldfile> <somedirectory>
- Move file: mv
 - Can be also thought of as rename
 - Works like cp
- Show file contents: cat
 - Often used to concatenate files, hence the name

Common Shell Commands

- Show file contents page by page: more
 - Superceded by: less
- Remove file: rm
 - Remove directory: rm -r
- Show the beginning of file: head
- Show the ending of file: tail
- Count file characters, words and lines: wc

Pipes and Redirection

- Pipe std out of command1 to command 2: command1 | command2
- Redirect std out of command to file: command>file
- Redirect file to std input: command<input
 - Can also do: cat file| command
- Example
 - cat file1 file2 | sort | less